Proceedings of the 2006 IEEE
International Conference on Control Applications
Munich, Germany, October 4-6, 2006

Real-time Validation and Comparison of Fuzzy Identification and State-space Identification for a UAV Platform

Shaaban A. Salman, Vishwas R. Puttige, Sreenatha G. Anavatti

Abstract—**Unmanned aerial vehicles (UAVs) have been playing an increasingly important role in military and civilian applications. Identification of UAV model is an important process in the controller design. In this paper, identification of the attitude dynamics of UAV is investigated. Two different identification techniques for attitude dynamics of UAV are applied, verified and compared together. The first method is based on an error mapping approach, while the second one is based on fuzzy system approach. The main features of the two identification methods are discussed and compared. The identification algorithms are programmed onto the microcontroller and a real time validation was performed using the in-house developed Hardware in loop simulation (HIL) tool. The performance of both identification approaches is evaluated based on the flight data. Real time simulation results show that the fuzzy identification approach is better than error mapping approach.**

I. Introduction

WHEN the system to be modeled is Unmanned Aircraft vehicle (UAV), the model is generally a multiple input multiple output one, and the measurements are noisy. While significant progress has been made in identification of linear systems over the broad spectrum of aerospace applications, insufficient research has been performed to identify the nonlinear flight dynamics [1]. It has been recognized that the significant improvements of dynamic performance of current and new generation of advanced airplanes are possible if flight systems design integrates nonlinear analysis, control, and identification [2]. More specifically, identification plays a fundamental role in the design of guidance and control systems for UAVs as well as in a number of related applications, such as, for example, underwater vehicles [3, 4]. The application of system identification (ID) techniques to UAV attitude dynamics deals with the estimation of a number of unknown parameters that characterize its behaviour on the basis of flight data.

In this paper, a comparison of two identification methods recently proposed for nonlinear systems is carried out. The first method, State-space ID [2, 5-7] is based on error mapping approach, while the second one [8-10] is based on fuzzy system using back propagation technique. The platform is a small light weight, low cost model plane built in-house. The hardware developed for the identification technique is a PIC microcontroller based autopilot unit with a 3 axis Inertial Measurement Unit (IMU), the entire unit weighing less than 400 grams. The real time validation of the identification algorithms is carried out by programming the logics into the micro-controller. The flight data is collected by the flight tests conducted at UNSW@ADFA.

The rest of the paper is organized as follows. Section 2 gives a brief description of the state space identification approach whereas section 3 gives a brief description of the fuzzy system identification approach. In section 4, real time validation is described using HIL and in section 5 some concluding remarks are presented.

Manuscript received February 10, 2006.

Dr. Sreenatha G Anavatti is a senior lecturer at the School of ACME at the Australian Defence Force Academy (UNSW@ADFA), Canberra, ACT-2600, Australia (corresponding author: ph: 61 2 62688079; fax: 61 2 62688276; e-mail: agsrenat@adfa.edu.au).

Shaaban A. Salman, is a PhD student at the School of ACME, UNSW@ADFA (e-mail: s.salman@adfa.edu.au).

Vishwas R. Puttige is a PhD student at the School of ACME at UNSW@ADFA (e-mail: v.puttige@adfa.edu.au).

II. State Space Identification

The nonlinear mapping identification method [2, 5-7] considers the system in the form

$$\dot{x}(t) = F(x,u)\,,\, t \ge 0\,,\, x(t_0) = x_0\,, \qquad (1)$$

Here $x \in R^c$ is the vector consisting of the measured states with initial conditions $x(t_0) = x_0$, $u \in R^m$ is the known input vector, and $F(x,u)$ denotes a continuous vector function which is defined on $R^c \setminus \{0\}$ with $F(0,0) = 0$.

System (1) can be written in the matrix state space form as [5].

$$\dot{x}(t) = A(t) f(x,u)\,,\, t \ge 0\,,\, x(t_0) = x_0\,, \qquad (2)$$

where $A(t) \in R^{cxn}$ is the real matrix, $f(x,u)$ denotes a given real analytic function, and $f(\bullet): R^c x R^m \to R^n$.

The identified state space model is defined as

$$\dot{x}_m(t) = A_m(t) f(x_m,u)\,,\, t \ge 0\,,\, x_m(t_0) = x_{m0}\,. \qquad (3)$$

Matrix coefficients, $A_m(t)$, are to be identified from flight data.

The normalized parameter error matrix $\Delta A(t) \in R^{cxn}$ is defined as

$$\Delta A(t) = A(t) - A_m(t)\,. \qquad (4)$$

The state error vector is introduced as

$$\begin{aligned}\Delta\dot{x}(t) &= \dot{x}(t) - \dot{x}_m(t)\\ &= \Delta A(t) f(x,u) + A_m(t)\Delta f(x,x_m,u),\\ t \ge 0,\ &\Delta x(t_0) = \Delta x_0.\end{aligned}$$

Here, $\Delta f(x, x_m, u) = f(x,u) - f(x_m, u)$.

The error vector is defined as

$$e(t) = \Delta\dot{x}(t) - A_m(t)\Delta f(x, x_m, u) = \Delta A(t) f(x,u) \quad (5)$$

An identification algorithm converges if

$\lim_{t\to+\infty} \|\Delta A(t)\| = 0$ and hence $\lim_{t\to+\infty} \|e(t)\| = 0$

Using the differential equation for the normalized parameter error matrix as in [7], one has

$$\Delta\dot{A}(t) = -e(t) f(x,u)^T K, K \in R^{nxn}, \Delta A(t_0) = \Delta A_0 \quad (6)$$

where K is the weighting matrix. The selection of K affects the convergence of the identified parameters to their real values. It is chosen by the designer to guarantee the convergence and to attain the desired convergence rate.

From (5) and (6), one gets

$$\dot{A}_m(t) = \dot{A}(t) + e(t) f(x,u)^T K, A_m(t_0) = A_{m0} \quad (7)$$

At a given flight condition, UAV dynamics can be assumed as time invariant, i.e., system parameters are constant. Hence system (2) becomes time invariant, so $\dot{A}(t) = 0$.

Then, we have the following nonlinear equation

$$\begin{aligned} &\dot{A}_m(t) = [\Delta\dot{x}(t) - A_m \Delta f(x, x_m, u)] f(x,u)^T K, \\ &A_m(t_0) = A_{m0} \end{aligned} \quad (8)$$

The unknown parameters are found by solving nonlinear differential equation (8).

In this paper, the attitude dynamics of UAV is considered. The attitude dynamics of the UAV is mapped by a set of three highly coupled nonlinear differential equations

$$\begin{aligned} \dot{p} &= \frac{1}{I_x I_z - I_{xz}^2}\begin{Bmatrix} I_z[L + (I_y - I_z)qr] + \\ I_{xz}[N + (I_x - I_y + I_z)pq - I_{xz}qr] \end{Bmatrix} \\ \dot{q} &= \frac{1}{I_y}[M + pr(I_z - I_x) + (r^2 - p^2)I_{xz}] \\ \dot{r} &= \frac{1}{I_x I_z - I_{xz}^2}\begin{Bmatrix} I_x[N + (I_x - I_y)pq] + \\ I_{xz}[L + (I_y - I_x - I_z)qr + I_{xz}pq] \end{Bmatrix} \end{aligned} \quad (9)$$

where $p(t), q(t)$, and $r(t)$ are the roll, pitch and yaw rates respectively, I_x, I_y, and I_z are the moment of inertia about x, y, and z respectively, I_{xz} is the product moment of inertia, $\delta_e, \delta_r, \delta_a$, and δ_{th} are the elevator, rudder, aileron and throttle servo deflections respectively and *L, M* and *N's* are the roll, pitch and yaw aerodynamic moments.

Modeling the aerodynamics moments as

$$L = L(p, r, \delta_a, \delta_r, \delta_{th}) = L_p p + L_r r + L_{\delta_a}\delta_a + L_{\delta_r}\delta_r + L_{\delta_{th}}\delta_{th}$$

$$M = M(q, \delta_e, \delta_{th}) = M_q q + M_{\delta_e}\delta_e + M_{\delta_{th}}\delta_{th}$$

$$N = N(p, r, \delta_a, \delta_r, \delta_{th}) = N_p p + N_r r + N_{\delta_a}\delta_a + N_{\delta_r}\delta_r + N_{\delta_{th}}\delta_{th}$$

The state vector and input vector are given by

$x(t) = [p \quad q \quad r]^T = [x_1 \quad x_2 \quad x_3]^T$ and

$u(t) = [\delta_e \quad \delta_r \quad \delta_a \quad \delta_{th}]^T = [u_1 \quad u_2 \quad u_3 \quad u_4]^T$ respectively.

By choosing

$$\begin{aligned} f(x,u) &= [pq\ qr\ pr\ p^2\ r^2\ p\ q\ r\ \delta_e\ \delta_r\ \delta_a\ \delta_{th}]^T \\ &= [x_1x_2\ x_2x_3\ x_1x_3\ x_1^2\ x_3^2\ x_1\ x_2\ x_3\ u_1\ u_2\ u_3\ u_4]^T \end{aligned}$$

and rewriting (9) according to (2), one obtains

$$A(t) = \begin{bmatrix} A_{11} & 0 & A_{31} \\ A_{12} & 0 & A_{32} \\ 0 & A_{23} & 0 \\ 0 & A_{24} & 0 \\ 0 & A_{25} & 0 \\ A_{16} & 0 & A_{36} \\ 0 & A_{27} & 0 \\ A_{18} & 0 & A_{38} \\ 0 & A_{29} & 0 \\ A_{110} & 0 & A_{310} \\ A_{111} & 0 & A_{311} \\ A_{112} & A_{212} & A_{312} \end{bmatrix}^T$$

where,

$A_{11} = \dfrac{I_{xz}(I_x - I_y + I_z)}{I_x I_z - I_{xz}^2}$, $A_{12} = \dfrac{I_z(I_y - I_z) - I_{xz}^2}{I_x I_z - I_{xz}^2}$,

$A_{16} = \dfrac{I_z L_p + I_{xz} N_p}{I_x I_z - I_{xz}^2}$, $A_{18} = \dfrac{I_z L_r + I_{xz} N_r}{I_x I_z - I_{xz}^2}$,

$A_{110} = \dfrac{I_z L_{\delta_r} + I_{xz} N_{\delta_r}}{I_x I_z - I_{xz}^2}$, $A_{111} = \dfrac{I_z L_{\delta_a} + I_{xz} N_{\delta_a}}{I_x I_z - I_{xz}^2}$,

$A_{112} = \dfrac{I_z L_{\delta_{th}} + I_{xz} N_{\delta_{th}}}{I_x I_z - I_{xz}^2}$

$A_{23} = \dfrac{I_z - I_x}{I_y}$, $A_{24} = \dfrac{-I_{xz}}{I_y}$, $A_{25} = \dfrac{I_{xz}}{I_y}$,

$A_{27} = \dfrac{M_q}{I_y}$, $A_{29} = \dfrac{M_{\delta_e}}{I_y}$, $A_{212} = \dfrac{M_{\delta_{th}}}{I_y}$,

$A_{31} = \dfrac{I_x(I_x - I_y) + I_{xz}^2}{I_x I_z - I_{xz}^2}$, $A_{32} = \dfrac{I_{xz}(I_y - I_x - I_z)}{I_x I_z - I_{xz}^2}$,

$A_{36} = \dfrac{I_x N_p + I_{xz} L_p}{I_x I_z - I_{xz}^2}$, $A_{38} = \dfrac{I_x N_r + I_{xz} L_r}{I_x I_z - I_{xz}^2}$,

$A_{310} = \dfrac{I_x N_{\delta_r} + I_{xz} L_{\delta_r}}{I_x I_z - I_{xz}^2}$, $A_{311} = \dfrac{I_x N_{\delta_a} + I_{xz} L_{\delta_a}}{I_x I_z - I_{xz}^2}$,

and $A_{312} = \dfrac{I_x N_{\delta_{th}} + I_{xz} L_{\delta_{th}}}{I_x I_z - I_{xz}^2}$

The matrix coefficients A are unknown and need to be identified.

The model identifier equation (3) is given by

$$\dot{x}_{m1} = A_{m11}x_{m1}x_{m2} + A_{m12}x_{m2}x_{m3} + A_{m16}x_{m1} + A_{m18}x_{m3} + A_{m110}u_2 + A_{m111}u_3 + A_{m112}u_4$$
$$\dot{x}_{m2} = A_{m23}x_{m2}x_{m3} + A_{m24}x^2{}_{m1} + A_{m25}x^2{}_{m3} + A_{m27}x_{m2} + A_{m29}u_1 + A_{m212}u_4$$
$$\dot{x}_{m3} = A_{m31}x_{m1}x_{m3} + A_{m32}x_{m2}x_{m3} + A_{m36}x_{m1} + A_{m38}x_{m3} + A_{m310}u_2 + A_{m311}u_3 + A_{m312}u_4$$

Equation (5) leads to

$$\begin{aligned} e_1(t) &= \Delta\dot{x}_1 - [A_{m11}(x_{m1}x_{m3} - x_1x_3) + A_{m12}(x_{m2}x_{m3} - x_2x_3) \\ &\quad + A_{m16}(x_{m1} - x_1) + A_{m18}(x_{m3} - x_3)] \\ e_2(t) &= \Delta\dot{x}_2 - [A_{m23}(x_{m1}x_{m3} - x_1x_3) + A_{m24}(x^2{}_{m1} - x^2{}_1) \\ &\quad + A_{m25}(x^2{}_{m3} - x^2{}_3) + A_{m27}(x_{m2} - x_2)] \\ e_3(t) &= \Delta\dot{x}_3 - [A_{m31}(x_{m1}x_{m3} - x_1x_3) + A_{m32}(x_{m2}x_{m3} - x_2x_3) \\ &\quad + A_{m36}(x_{m1} - x_1) + A_{m38}(x_{m3} - x_3)] \end{aligned} \tag{10}$$

Using equations (7),(8) and (10) , one gets

$$\begin{aligned} \dot{A}_m(t) &= ef(x,u)^T K \\ &= \left[e_1(t)\, e_2(t)\, e_3(t)\right] \\ &\left[x_1x_2\ x_2x_3\ x_1x_3\ x_1^2\ x_3^2\ x_1\ x_2\ x_3\ \delta_e\ \delta_a\ \delta_r\ \delta_{th}\right]K \end{aligned}$$

If we assume that K is a diagonal matrix then,

$$\begin{aligned} &\frac{dA_{m11}}{dt} = e_1(t)x_1(t)x_2(t)k_{11}, \frac{dA_{m12}}{dt} = e_1(t)x_2(t)x_3(t)k_{22}, \\ &\frac{dA_{m16}}{dt} = e_1(t)x_1(t)k_{66}, \frac{dA_{m18}}{dt} = e_1(t)x_3(t)k_{88}, \\ &\frac{dA_{m110}}{dt} = e_1(t)u_2(t)k_{1010}, \frac{dA_{m111}}{dt} = e_1(t)u_3(t)k_{1111}, \\ &\frac{dA_{m112}}{dt} = e_1(t)u_4(t)k_{1212}, \frac{dA_{m23}}{dt} = e_2(t)x_2(t)x_3(t)k_{33}, \\ &\frac{dA_{m24}}{dt} = e_2(t)x_1^2(t)k_{44}(14a), \frac{dA_{m25}}{dt} = e_2(t)x_3^2(t)k_{55}, \\ &\frac{dA_{m27}}{dt} = e_2(t)x_2(t)k_{77}, \frac{dA_{m29}}{dt} = e_2(t)u_1(t)k_{99}, \\ &\frac{dA_{m212}}{dt} = e_2(t)u_4(t)k_{1212}, \frac{dA_{m31}}{dt} = e_3(t)x_1(t)x_2(t)k_{11}, \\ &\frac{dA_{m32}}{dt} = e_3(t)x_2(t)x_3(t)k_{22}, \frac{dA_{m36}}{dt} = e_3(t)x_1(t)k_{66}, \\ &\frac{dA_{m38}}{dt} = e_3(t)x_3(t)k_{88}, \frac{dA_{m310}}{dt} = e_3(t)u_2(t)k_{1010}, \\ &\frac{dA_{m311}}{dt} = e_3(t)u_3(t)k_{1111}, \frac{dA_{m312}}{dt} = e_3(t)u_4(t)k_{1212} \end{aligned} \tag{11}$$

By solving nonlinear equation (11) the unknown parameters can be identified.

The weighting matrix K is chosen so that the convergence of the identified parameters to the actual values is very fast. For the present work, it was selected as

$$K = diag[k_{ij}] \in R^{12x12}, \text{ with } k_{ij} = 0.001.$$

The idea behind choosing K matrix as diagonal is to make each coefficient of matrix A_m (as example A_{m11}, A_{m12} and ...etc) only affected by the corresponding multiplying terms (as example x_1x_2, x_2x_3 and ...etc). Hence only the weight of the error in coefficient of term will contribute in the correction of the coefficient. Also to check for the practicality of the K matrix, it was used at different flight conditions. However the results for a particular flight condition only are presented in this paper.

III. Fuzzy Identification

A fuzzy system performs a mapping from $U \subset \mathrm{R}^n$ to R, where $U = U_1 x ... U_n$, $U_i \subset R$, and $i = 1,2,...,n$. This mapping is performed based on another mapping between fuzzy sets [11].

A fuzzy set F of a universe of discourse U is characterized by a membership function $\mu_F : U \to [0,1]$, which associates with each element u of U, a number $\mu_F(u)$ in the interval $[0,1]$. The $\mu_F(u)$ represents the grade of membership of u in F. The label F of a fuzzy set is often some linguistic term like “small,” “large,” etc. There are four principal parts in a fuzzy system: fuzzifier, fuzzy rule base, fuzzy inference engine, and defuzzifier. The *fuzzy rule base* consists of a collection of fuzzy If–Then rules [9], [12] in the following form:

If x_1 is F_1^l and … and x_n is F_n^l

Then y is G^l

where $\underline{x} = (x_1, x_2, ..., x_n)^T \in U$ and $y \in R$ are the input and output of the fuzzy system, respectively, F_i^l and G^l are labels of fuzzy sets in $U_i (i = 1,2,...,n)$ and R, respectively, and $l = 1,2,..., M$, where M is the total number of fuzzy If–Then rules in the rule base. The *fuzzy inference engine* is the decision making logic, which employs fuzzy rules from the fuzzy rule base to determine a mapping from the fuzzy sets in U to the fuzzy sets in R. The *fuzzifier* maps a crisp point $\underline{x} = (x_1, x_2, ..., x_n)^T \in U$ into a fuzzy set defined on U. It provides input fuzzy sets for the fuzzy inference engine. There are (at least) two possible choices for the fuzzifier: the *singleton fuzzifier*, and the *nonsingleton fuzzifier* [13]. The most widely used one is the singleton fuzzifier, mainly because of its simplicity and lower computational requirements. The *defuzzifier* maps fuzzy sets in R to crisp points in R. The most commonly used method is the *center-average defuzzifier* [14].

Fuzzy model identification is an effective tool for the approximation of uncertain nonlinear systems on the basis of measured data. The identification of a fuzzy model using input-output data can be divided into two tasks [15], [16], [17]: structure identification, which determines the type and number of the rules and membership functions, and parameter identification which adjust fuzzy system parameters such as membership parameters. For both structural and parametric adjustment, prior knowledge plays an important role. Hence, in the present work the rules of the fuzzy system are designed based on the available prior knowledge and the parameters of the membership. The consequent functions are adapted in a learning process based on the available input-output data. A multiple input multiple output system can be considered as several multiple inputs single output systems.

In the present work UAV attitude dynamic is the main concern, and four inputs and three outputs are involved. Hence, UAV attitude dynamics is considered as three separate multiple inputs single output systems. The first one is the pitch fuzzy system, where the inputs are the elevator ($\delta_e(k)$) and throttle ($\delta_{th}(k)$) deflections and the output is the pitching rate *q(k)*. The second one is the roll fuzzy system where the inputs are the rudder ($\delta_r(k)$), aileron ($\delta_a(k)$) and throttle ($\delta_{th}(k)$) deflections and the output is the roll rate *p(k)*. The last one is the yaw fuzzy system where the inputs are the rudder ($\delta_r(k)$), aileron ($\delta_a(k)$) and throttle ($\delta_{th}(k)$) deflections and the output is the yaw rate $r(k)$. The Gaussian membership function is used for all inputs where

$\mu_f(x_i) = a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)$, where x_i is the i[th] input crisp point to the fuzzy system, x_i^l, σ_i^l and a_i^l are the Gaussian membership parameters. In the present work, we assume that $a_i^l = 1$ because intuitively we can assume μ_f achieves 1 at some points.

Using the product inference rule and singleton fuzzifier with center average defuzzifier one obtains

$$f(x) = \frac{\sum_{l=1}^{M} y^l \left[\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)\right]}{\sum_{l=1}^{M} \left[\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)\right]} \tag{12}$$

where y^l is point at which the degree of membership function achieves its maximum value one at this point. M and n are the number of rules and inputs respectively.

The purpose of the fuzzy identification is that given an input-output data pair (x, d) where $x \in U \subset R^n$ and $d \in V \subset R$, determine a fuzzy logic system $f(x)$ such that

$$e = \frac{1}{2}\left[f(x) - d\right]^2 \tag{13}$$

is minimized, where $f(x)$ is the model output and d is the real output.

Three parameters, y^l, x_i^l and σ_i^l need to be adjusted. The training procedure for y^l is:

$$y^l(k+1) = y^l(k) - \alpha \left.\frac{\partial e}{\partial y^l}\right|_k \tag{14}$$

$$\frac{\partial e}{\partial y^l} = (f - d)\frac{\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)}{\sum_{l=1}^{M}\left[\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)\right]} \tag{15}$$

where α is the step size used.

To train x_i^l, the following adaptive rules are used

$$x_i^l(k+1) = x_i^l(k) - \alpha \left.\frac{\partial e}{\partial x_i^l}\right|_k \tag{16}$$

$$\frac{\partial e}{\partial x_i^l} = (f - d)(y^l - f)\left(\frac{\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)}{\sum_{l=1}^{M}\left[\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)\right]}\right)\frac{2(x_i - x_i^l)}{\sigma_i^{l2}} \tag{17}$$

To train σ_i^l, the following adaptive rules are used

$$\sigma_i^l(k+1) = \sigma_i^l(k) - \alpha \left.\frac{\partial e}{\partial \sigma_i^l}\right|_k \tag{18}$$

$$\frac{\partial e}{\partial \sigma_i^l} = (f - d)(y^l - f)\left(\frac{\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)}{\sum_{l=1}^{M}\left[\prod_{i=1}^{n} a_i^l \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l}\right)^2\right)\right]}\right)\frac{2(x_i - x_i^l)^2}{\sigma_i^{l3}} \tag{19}$$

The training algorithms (14), (16), and (18) perform an error back propagation procedure.

In this paper three fuzzy systems are employed as described earlier. Three rules for each fuzzy system are employed. Also the space partitioning for the inputs and outputs depends on the range of a collected data.

IV. Real-time Hardware In the Loop Simulations

'Real-time Hardware in the Loop' simulation has been developed for the testing of the identification algorithm before flying the UAV. Here the Autopilot unit is subjected to a similar situation as it would experience during the actual flight. Different universities have come up with their own versions of validation of control techniques [18-20]. The major drawbacks in most of these validation techniques are either they are cumbersome to implement with complicated operating systems or were not implemented on a real-time platform. The validation technique used here is a simple testbed developed in-house using the Serial port data communication feature. The two major software platforms for this virtually created environment are the real-time workshop on the Matlab [21] and the MPLAB Development environment supplied by Microchip Inc [22]. The simulation developed here is accurate up to the second decimal place. The HIL simulation is divided into 3 sections:

Terminal A: The Matlab/Simulink platform: Some of the toolboxes in Simulink proved to be effective in the development of the HIL simulation. Matlab/Simulink is used to transmit previously collected flight data values to the Autopilot unit for Identification. The aircraft model developed transmits the sensor information to the microcontroller and receives the servo data from it.

Terminal B: MPLAB IDE (Integrated Development Environment) platform: The PIC 18F8720 which is the main microcontroller has two Universal Synchronous Asynchronous Receive and Transmit (USART) ports. The Asynchronous communication with full duplex data transmission is being achieved here for the simulation. A RS-232 driver converts the microcontroller output to a RS-232 compatible voltage level and transmits it at a specified baud rate. The entire identification algorithm is programmed in Microchip C18 programming language compiler running on the MPLAB development environment. The microcontroller accepts sensor data, processes it and transmits the servo data or the identified data to the Matlab in real-time.

The serial interface: The communication between the two terminals is achieved through the serial RS-232 ports available both on the Autopilot unit and the computer running Matlab. The medium chosen for this is a 25 pin to 9 pin serial RS-232 cable. Data is transmitted and received at a fixed baud rate at both ends.

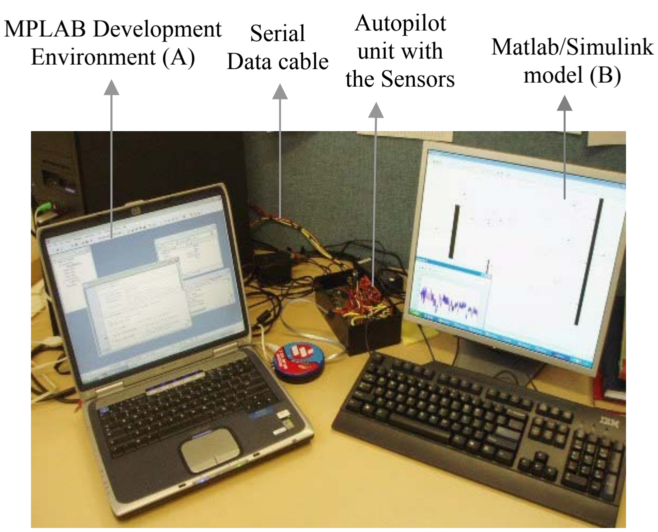

Fig. 1. HIL simulation setup with the Autopilot

Figures 1 and 2 show a simplified picture of the HIL simulation. The microcontroller transmits data to the Simulink model in real time using the serial communication feature. The aircraft model in Simulink receives data on the serial channel formulates the next set of outputs and transmits it out to the microcontroller.

The comparison tests were performed on six different flight data sets taken during different test flights. Figures (3) and (4) show the outputs of the identified model (bold) and the actual flight (gray) from the HIL simulation for the fuzzy model and the statespace model explained the previous sections. Table 1 (included at the end of the paper) shows the average error for the two methods. As shown in these figures and table 1, fuzzy model gives a better result than the statespace model.

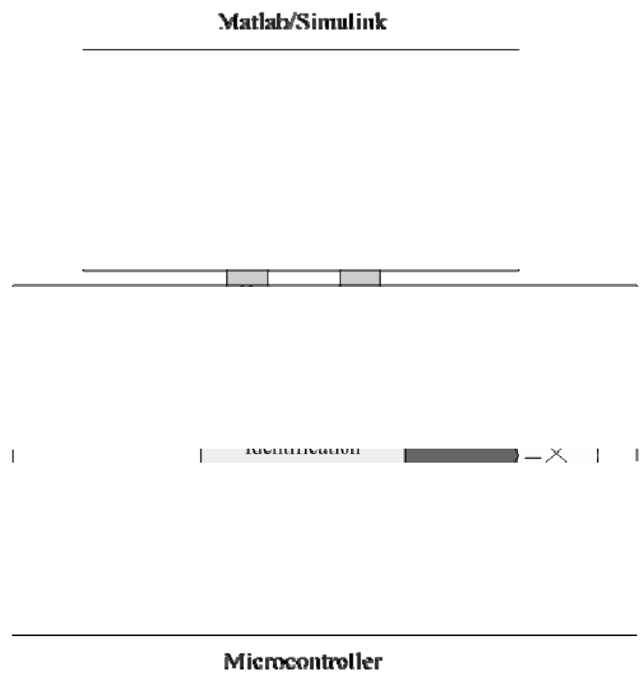

Fig. 2. Block diagram of the Hardware in Loop simulation

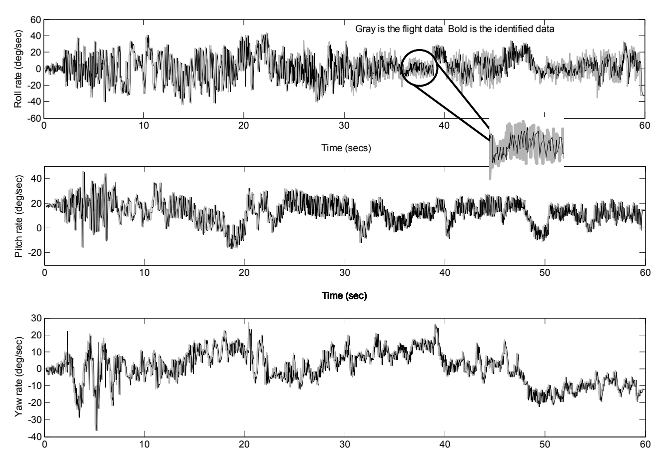

Fig. 3. Fuzzy identified model (black) and flight data (gray)

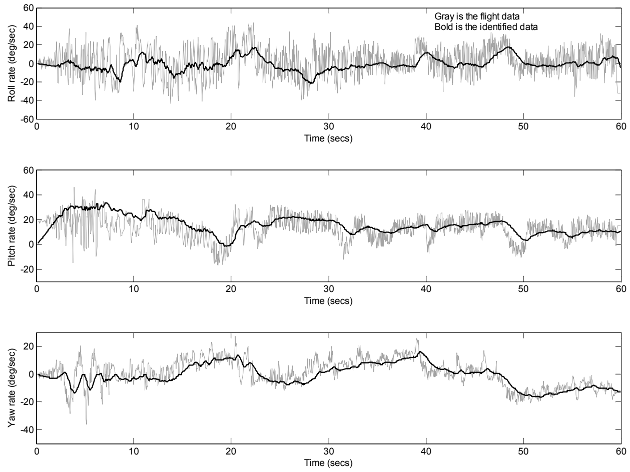

Fig. 4. State space identified model (black) and flight data (gray)

V. CONCLUSION

In this paper, two different online identification techniques are presented. The complexity and difficulties of state space identification method is in model structure selection which needs an engineering intuition combined with a prior knowledge of the system behavior. Alternatively the fuzzy system only needs to be trained to learn the real non-linear relationships between inputs and outputs of the system under study. The only information required for training of fuzzy system is the input/output data. It means that there is no need for any prior knowledge of the physical relationship inside the system and it offer a ‘black box’ modeling tool. The state space identification method has the advantage of incorporating coupling between roll, pitch and yaw rates, while the fuzzy method does not. Real time simulation validations using hardware in loop simulation show good match for both the algorithms. This analysis was performed for six sets of data and fuzzy system proved to be better in all the cases. The results with the fuzzy system are relatively better in imitating the data as indicated by the reduced average square error. The design of suitable controllers based on the identified model is currently under development.

TABLE I
AVERAGE SQUARE ERROR FOR THE TWO METHODS

variable	Average square error Statespace method		Average square error for Fuzzy method	
	Noisy date	Filtered data	Noisy date	Filtered data
p	0.0964	0.0842	0.0481	0.0409
q	0.0255	0.0197	0.0149	0.0113
r	0.0118	0.0097	0.0085	0.0076

REFERENCES

[1] Morelli, E.A., “System Identification Programs for Aircraft (SIDPAC)," AIAA Atmospheric Flight Mechanics Conference and Exhibit, Monterey, California, Aug. 5-8, 2002.

[2] Lyshevski, S.E., "State-Space Identification of Nonlinear Flight Dynamics,” IEEE International Conference on Control Applications, Hartford, CT, 1997.

[3] Tiano, A., "Comparison of Non Linear Identification Methods For Underwater Vehicle," IEEE First International Symposium on Control, Communications and Signal Processing, 2004, pp. 549 - 552

[4] Leonessa, A. and Luo, D.," Nonlinear Identification of Marine Thruster Dynamics," MTS/IEEE Conference and Exhibition, Vol.1, 2001, pp. 501 - 507

[5] Lyshevski, C. and Yaobin Chen, "Nonlinear Identification of Aircraft," IEEE International Conference on Control Applications, Dearborn, MI, 1996, pp. 327 - 331.

[6] Pappano, V., Lyshevski, S.E., and Friedland, B., "Nonlinear Identification of Induction Motor Parameters," Proceeding of the American Control Conference, Vol. 5, San Diego, California, 1999, pp.3569 - 3573.

[7] Lyshevski S.E., "Identification of Nonlinear Flight Dynamics: Theory and Practice," IEEE Transactions On Aerospace and Electronic Systems, Vol. 36, Issue 2, 2000, pp.383 - 392.

[8] Wang, L.-X, "Adaptive Fuzzy Systems and Control: Design and Stability Analysis,” prentice Hall, Inc., 1994.

[9] Wang, L.-X., "Design and Analysis of Fuzzy Identifiers of Nonlinear Dynamic Systems,” IEEE Transactions on Automatic Control, Vol. 40, Issue 1, Jan. 1995, pp.11 - 23.

[10] Wang, R.L., "Complex Systems Modeling Via Fuzzy Logic," IEEE Trans. Systems Man Cybernet, Vol.26, issue 1, 1996, pp. 100–106.

[11] Zadeh, L.A., "Fuzzy Logic," Computer, IEEE, Vol. 21, Issue 4, 1988, pp.83 - 93.

[12] Hojati, M. and Gazor, S., "Hybrid Adaptive Fuzzy Identification and Control of Nonlinear Systems," IEEE Transactions on Fuzzy Systems, Vol. 10, Issue 2, April 2002, pp198 - 210.

[13] Mouzouris, G.C. and Mendel, J.M.," Nonsingleton Fuzzy Logic Systems: Theory And Application," IEEE Transactions on Fuzzy Systems, Vol. 5, Issue 1, Feb. 1997, pp.56 - 71.

[14] Lee, C.C. "Fuzzy Logic in Control Systems: Fuzzy Logic Controller-Part 1," IEEE Transactions On Systems, Man, And Cybernetic's, Vol. 20, Issue 2, March-April 1990, pp. 404 - 418.

[15] Emami, M.R., Turksen, I.B. and Goldenberg, A.A, "Development of A Systematic Methodology of Fuzzy Logic Modeling,” IEEE Transactions On Fuzzy Systems, Vol.6, No. 3, August, 1998, pp. 346-361.

[16] Akkizidis, I. S. and Roberts, G. N., "Fuzzy Clustering Methods for identifying and modelling of non linear control strategies," Proceedings of the I MECH E Part I Journal of Systems & Control Engineering, Vol. 215, Number 5, October 2001, pp. 437-452.

[17] Salehfar, H.,Bengiamin, N. and Huang, J., "A Systematic Approach To Linguistic Fuzzy Modeling Based On Input-Output Data," Winter Simulation Conference, Orlando, Florida, USA, 2000, pp. 48-486.

[18] Jung Soon Jang and Claire J. Tomlin "Autopilot Design for the Stanford DragonFly UAV: Validation through Hardware-in-the-Loop," AIAA Guidance, Navigation, and Control Conference and Exhibit, Montreal, Canada, Aug. 6-9, 2001.

[19] Johnson, E. and Mishra , S., "Flight Simulation For The Development Of An Experimental UAV," AIAA Modeling and Simulation Technologies Conference and Exhibit, Monterey, California, Aug. 5-8, 2002.

[20] Simon Fürst, S.W., Ernst D. Dickmanns. "A Single-Computer HWIL Simulation Facility For Real-Time Vision Systems," SPIE Proceedings Vol. 3368, 1998.

[21] Extended Real Time Toolbox For Matlab Simulink Model, 2005.

[22] Microchip Technology Inc. 2005.